Comparison of Different Methods for Next Location Prediction

Jan Petzold, Faruk Bagci, Wolfgang Trumler, and Theo Ungerer

Institute of Computer Science, University of Augsburg,
Eichleitnerstr. 30, 86159 Augsburg, Germany
{petzold, bagci, trumler, ungerer}@informatik.uni-augsburg.de

Abstract. Next location prediction anticipates a person's movement based on the history of previous sojourns. It is useful for proactive actions taken to assist the person in an ubiquitous environment. This paper evaluates next location prediction methods: dynamic Bayesian network, multi-layer perceptron, Elman net, Markov predictor, and state predictor. For the Markov and state predictor we use additionally an optimization, the confidence counter. The criterions for the comparison are the prediction accuracy, the quantity of useful predictions, the stability, the learning, the relearning, the memory and computing costs, the modelling costs, the expandability, and the ability to predict the time of entering the next location. For evaluation we use the same benchmarks containing movement sequences of real persons within an office building.

1 Introduction

Can the movement of people working in an office building be predicted based on room sequences of previous movements? In our opinion people follow some habits, but interrupt their habits irregularly, and sometimes change their habits. Moreover, moving to another office fundamentally changes habits too. Thus location prediction methods need to exhibit some features: high prediction accuracy, a short training time, retention of prediction in case of irregular habitual interrupts, but an appropriate change of prediction in case of habitual changes.

Location predictions with such features could be used for a number of applications in ubiquitous and mobile environments.

- Smart doorplates that are able to direct visitors to the current location of an office owner based on a location-tracking system and predict if the office owner is soon coming back [14].
- Similarly, next location prediction within a smart building can be used to prepare the room which is presumably entered next by a habitant, e.g. by phone call forwarding.
- Outdoor movement patterns can be used to predict the next region a person will enter.
- Elevator prediction could anticipate at which floor an elevator will be needed next.

– Routing prediction for cellular phone systems may predict the next radio cell a cellular phone owner will enter based on his previous movement behaviour.

We considered the first application in more detail and used benchmarks with movement data of four persons over several months. The benchmarks are called Augsburg Indoor Location Tracking Benchmarks. They are publicly available [9], and are applied to evaluate several prediction techniques and to compare the efficiency of these techniques with exactly the same evaluation set-up and data.

Our aim is to investigate how far machine learning techniques can dynamically predict room sequences and time of room entry independent of additional knowledge. Of course the information could be combined with contextual knowledge as e.g. the office time table or personal schedule of a person, however, in this paper we focus on dynamic techniques without contextual knowledge.

Time of arrival at the predicted location depends on the sojourn time at the current location plus the rather constant time to move to the predicted location. The sojourn time was modelled into the presented Bayesian network. We tested also a time prediction which calculated the mean and the median of the previous sojourn times within a location. The best results were reached by the median. The time prediction is independent of the location prediction method and can easily be combined with any of the regarded methods. Therefore we restrict this comparison to location prediction only.

Several prediction techniques are proposed in literature — namely Bayesian networks, Markov models or Hidden Markov models, various neural network approaches, and the state predictor methods. The challenge is to transfer these algorithms to work with context information. In this paper we choose five approaches, a dynamic Bayesian network, a multi-layer perceptron, an Elman net, a Markov predictor, and a state predictor. In the case of the Markov predictor and the state predictor we use additionally a version which is optimized by confidence estimation.

There are a lot of methodological problems for a fair comparison of such diverse methods. The models are different and hard to compare. We chose the same set-up to model all methods and for each method the best model that we could find. Moreover we had the choice either to combine all persons within a single model thus potentially making improvements by detecting correlations between person movements or to model each person separately. We chose the latter simpler model because simulations with the combined model using the Augsburg Benchmarks showed no improvements.

The main criterion for comparison is the average prediction accuracy of the different methods. Another question concerns the model and the modelling costs of the technique. Which parameters exist and influence the model? What happens if one parameter is changing? We call this the stability of the techniques. Can the model simply be extended by more or other locations? The answer to this question allows to assess how well the model can be transferred to other applications.

Further interesting questions concern the efficiency of training of a predictor, before the first useful predictions can be performed, and of retraining, i.e. how long it takes until the predictor adapts to a habitual change and provides again useful predictions. Predictions are called useful if a prediction is accurate with a certain confidence level. Moreover, memory and performance requirements of a predictor are of interest in particular for mobile appliances with limited performance ability and power supply.

The next section states related work on context prediction. Section 3 introduces shortly the five approaches and the applied location models. For detailed information about the basic techniques use the stated references. Section 4 gives the evaluation results. The paper ends with the conclusions.

2 Related Work

The Adaptive House project [7] of the University of Colorado developed a smart house that observes the lifestyle and desires of the inhabitants and learned to anticipate and accommodate their needs. Occupants are tracked by motion detectors and a neural network approach is used to predict the next room the person will enter and the activities he will be engaged. Patterson et al. [8] presented a method of learning a Bayesian model of a traveller moving through an urban environment based on the current mode of transportation. The learned model was used to predict the outdoor location of the person into the future.

Markov chains are used by Kaowthumrong et al. [5] for active device selection. Ashbrook and Starner [1] used location context for the creation of a predictive model of user's future movements based on Markov models. They propose to deploy the model in a variety of applications in both single-user and multi-user scenarios. Their prediction of future location is currently time independent, only the next location is predicted. Bhattacharya and Das [2] investigate the mobility problem in a cellular environment. They deploy a Markov model to predict future cells of a user. An architecture for context prediction was proposed by Mayrhofer [6] combining context recognition and prediction. Active LeZi [4] was proposed as good candidate for context prediction.

There are several publication of our group which present next location prediction in an office building. In [10] we proposed the basic state predictor technique which is similar to the Markov predictor, but an automaton is used for the prediction. In [11] an enhancement by confidence estimation techniques is presented. Vintan et al. [15] applied a multi-layer perceptron and Petzold et al. [12] proposed a dynamic Bayesian network to predict indoor movements of several persons.

The contribution of this paper is the comparison of five different prediction methods including the new Elman net approach and the confidence estimation applied to the Markov predictor. According to our knowledge no comparative studies of different methods with the same evaluation setups and benchmarks exist.

3 Prediction Methods

Figure 1 shows the next location prediction principle which is used by each investigated model. The input consists only of the sequence of the last visited locations and the entry time of these locations. The output is the possible next location and the appropriate entry time.

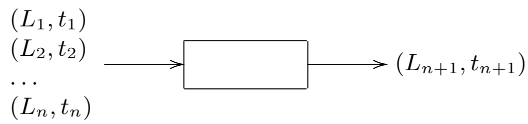

Fig. 1. Next location prediction

Dynamic Bayesian Network

In order to predict the next location of a person, a dynamic Bayesian network was chosen. Additionally the time is predicted when the person is probably entering the next location. In different simulations we looked for the best settings [12]. We detected that the prediction of next location is independent of the time parameter like the time of day and the weekday. Therefore we chose this proposed dynamic Bayesian network without these time dependencies for the comparison. As history we elected 2 for a better comparison based on similar memory costs.

Multi-Layer Perceptron

For next location prediction we chose the simplest multi-layer perceptron with one hidden layer and used a modified back-propagation algorithm for learning [15]. In principal each location would be represented by a single input and a single output neuron. However, we chose a binary encoding because it saves computing costs. This fact is interesting for mobile devices which must achieve some energy and real-time restrictions. The optimal parameter values for the network structure and the learning algorithm were determined by many simulation runs and are summarized in table 1.

Elman Net

The Elman net is another neural network method which expands the multi-layer perceptron by another hidden layer – the context layer. The context neurons provide storage for the activation states of the hidden neurons. This generates a dependency between two propagations within the net, since the hidden neurons get information from the input and the context neurons across the weighted connections to perform the next step. The number of the context neurons corresponds with the number of hidden neurons. To find the optimal parameters

Table 1. Optimal parameter values of the multi-layer perceptron

parameter	investigated values	optimal value
network structure		
history	[1;6]	2
number of hidden neurons	{5;7;...;15}	9
learning algorithm		
threshold	{0.1;0.3;...;0.9}	0.1
learning rate	[0.05;0.30]	0.10
number of backward steps	[1;5] and unlimited	1

of the net many simulations were performed. Table 2 shows the investigated and the optimal values of the parameters separated in parameter of the network structure and the learning algorithm. Since the Elman net is a recurrent network, the information about previous locations is modelled in the context cells. Therefore the history consists only of the current location.

Table 2. Optimal parameter values of the Elman net

parameter	investigated values	optimal values
network structure		
encoding	binary, one to one	one to one
number of hidden neurons	[5;20]	5
history	[1;5]	1
learning algorithm		
initialization	random, fix	fix
activation function	$\tanh(x)$, $\frac{1}{1+\exp(-x)}$	$\tanh(x)$
learning cycles	[5;150]	31
learning rate η	[0.1;0.7]	0.1
momentum α	[0.00;0.05;...;0.95]	0.00

Markov Predictor

Markov models seem a good approach for the next location prediction based on location histories. A Markov model regards a pattern of the last visited locations of a user to predict the next location. The length of the regarded pattern is called the order. Thus a Markov model with order 3 uses the last three visited locations. For all patterns the model stores the probabilities of the next location which is calculated from the whole sequence of the visited locations by the user. A simple Markov model is the Markov predictor [3, 13]. A Markov predictor stores for every pattern the frequencies of the next locations. For the comparison we chose an order of 2. Furthermore we will compare a Markov predictor which is optimized by confidence estimation [11].

State Predictor

A disadvantage of the Markov predictor is its bad relearning capability because of the frequency counter. After a habit change the new habit must be followed as often as the previous habit before the prediction is changed. The state predictors [10] prevent this problem. They use a finite automaton which is called two-state predictor for every pattern thus replacing the frequency counter of the Markov predictor. A state predictor with order 2 is used in the comparison.

The basic state predictor method can be significantly improved by some confidence estimation techniques [11]. One of the proposed methods, the confidence counter method, is independent of the used prediction algorithm. This method estimates the prediction accuracy with a saturation counter. Figure 2 shows a two-bit counter that consists of 4 states.

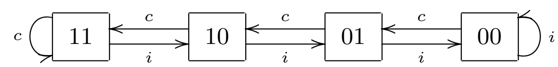

Fig. 2. Confidence counter

The initial state is state 10. Let s be the current state of the confidence counter. If a prediction result is proved as correct (c) the counter will be incremented, that means the state graph changes from state s into the state $s + 1$. If $s = 11$ the counter keeps the state s. Otherwise if the prediction is incorrect (i) the counter switches into the state $s - 1$. If $s = 00$ the counter keeps the state s. If the counter is in the state 11 or 10 the predictor is assumed as confident, otherwise the predictor is unconfident and the prediction result will not be supplied.

For the state predictor the prediction accuracy will be considered separately for every pattern. The confidence counter can also be applied with other techniques, in the evaluation a Markov predictor using the counter will be considered.

4 Evaluation

To evaluate the five techniques we chose the Augsburg Indoor Location Tracking Benchmarks taken from the Context Database of the University of Linz [9]. These benchmarks consist of two sets, the summer and the fall data. The used benchmarks contain the movements of four persons in an office building. The prediction accuracy is calculated for every person for all predictions from all rooms except the own office. For our comparison we tried to use models with similar memory costs. Therefore we didn't always choose the best setting for every technique. In fact we elected history 2 for the Markov predictor and the state predictor, which perform better with longer history but at the expense of large tables.

In the following we will compare all techniques on the basis of different criterions. The results are summarized in table 3.

Table 3. Comparison on the basis of the criterions

	Bayesian network	multi-layer perceptron	Elman net	Markov predictor	state predictor	Markov predictor with counter	state predictor with counter
accuracy (%) (quantity (%))	78.82 (89.89)	76.45 ($\approx$ 100)	79.68 (100)	76.53 (90.47)	70.89 (90.47)	81.14 (78.40)	81.88 (74.38)
stability (%)	29.67	32.59	71.57	24.67	29.97	24.95	23.99
learning	fast	slow	slow	fast	fast	fast	fast
relearning	slow	slow	slow	slow	fast	slow	fast
memory (bit)	6,500	3,880	7,215	36,960	2,730	37,380	3,150
computing costs	inefficient chain rule	training until $E < t$, otherwise one propagation	training over many learning cycles, otherwise one propagation	table look-up	table look-up	table look-up	table look-up
modelling costs	medium	high	high	low	low	low	low
expandability	yes	no	no	yes	yes	yes	yes
time prediction	integrated	parallel	parallel	parallel	parallel	parallel	parallel

Prediction accuracy. The prediction accuracy is calculated with the fall data; the summer data is used for the training. We assume that a prediction is needed after every location change. That means the number of requested predictions p is equal to the number of location changes. The Bayesian network, the Markov predictor and the state predictor cannot predict the next location if the current pattern occurs the first time. The number of predictions which cannot be delivered by these three techniques were denoted by p_n. In contrast the neural networks deliver a prediction if the code of the output vector corresponds to a location. Thus the Elman net predicts always a location ($p_n = 0$). Now we can determine the number of deliverable predictions $p_d = p - p_n$. Let c be the number of correct predictions then the prediction accuracy a is calculated as follows:

$$a = \frac{c}{p_d}$$

It isn't essential to make a prediction in our application, rather a prediction is an added value. Therefore we consider in the calculation of the accuracy only predictions which provide a result.

Table 3 shows the average prediction accuracy of the four persons for all techniques. If we consider the five techniques without confidence counter enhancement, the Elman net reaches the highest average prediction accuracy. If we consider the state predictor and Markov predictor using the confidence counter and compare them with all other methods, the state predictor with confidence counter delivers now the best accuracy.

Quantity. The number of deliverable predictions p_d is smaller than the number of requested predictions p. This gap can be determined by the quantity q:

$$q = \frac{p_d}{p}$$

The Elman net reaches a quantity of 100% because the net produces always an output vector. The quantity of the multi-layer perceptron is nearly 100% since not every code of the output vector corresponds with a location. The Bayesian network, the Markov predictor and the state predictor reach nearly the same quantity. With an optimization like the confidence estimation the quantity decreases.

Stability. The stability shows the impact of the change of a parameter. For the Bayesian network the history and the time parameters are a possibility to optimize the prediction accuracy. The multi-layer perceptron and the Elman net hold a multitude of parameters which can influence the prediction accuracy. Therefore the Elman net shows the worst stability. The Markov predictor and the state predictor give only the possibility to choose the order. Table 3 shows as stability the difference between the minimum and the maximum of the prediction accuracies reached with different parameters.

Learning. The learning phase of the neural networks takes a long time since the networks must be trained before they can be effectively used. The Bayesian network, the Markov predictor and the state predictor with and without confidence counter could make a prediction already after the second occurrence of a pattern.

Relearning. The neural networks, the Markov predictor without and with confidence counter need a long time for relearning. The Bayesian network relearns also slow. The state predictor with and without confidence counter relearns after two changes the new habit.

Memory costs. For the memory costs we calculated the minimal number of bits which will be needed to store the current state of the technique. For the evaluation all models were chosen to exhibit similar memory costs. In general, the memory costs of both neural networks are very low and independent from the number of location changes of a person. The Bayesian network needs only a small memory space, but the memory depends on the number of location changes. The state predictor requires the least memory. Against this the Markov predictor has the highest memory costs since the Markov predictor stores all frequencies. The costs of both predictors are dependent on the number of location changes. The confidence counter arises the costs insignificantly. Table 3 shows the memory costs for an upper limit of 500 location changes.

Computing costs. Because of the training process the computing costs of the neural networks are very high. The Elman net needs many learning cycles and the multi-layer perceptron will be trained until the error is less than a threshold.

The computing costs during the use of both neural networks are low because both execute only one backward propagation. The Bayesian network calculates the probabilities by the chain rule resulting in relatively high computation costs. The computing costs of the Markov predictor and the state predictor with and without the confidence counter consist of one table look-up.

Modelling efforts. For modelling the Bayesian network possible dependencies of the used variables must be extracted from the available data. Both neural networks require a high effort for modelling generated by the search for the optimal parameters. The costs for modelling the Markov predictor and the state predictor are low. A decision will only be needed concerning the length of the order. If a confidence counter is used the number of the counter states and the barrier must be determined.

Expandability. The expandability means the possibility to use the model with more locations. In the used benchmarks there are 15 locations. If we use a scenario with locations like the cells in a mobile network, the neural network models cannot be reused. A new modelling process with search for the optimal parameter is necessary. That means the neural network models cannot be reused without additional costs in another application. The Bayesian network, the Markov predictor and the state predictor with and without confidence counter can be expanded for more context without additional costs.

5 Conclusion

The paper compared five prediction techniques on the basis of different criterions. The comparison of the different techniques showed that there isn't an ultimative prediction technique. The user must decide which is the most important criterion for the application.

The Elman net reached the highest prediction accuracy, since it is a recurrent neural network which is affected by previous inputs. But both neural networks require high modelling costs, additional costs to expand for more contexts, and show the lowest stability. If the time prediction is the most important criterion the Bayesian network must be chosen. The state predictor should be applied if the prediction accuracy or the memory costs are the main facts. Compared to the Markov predictor, the state predictor relearns faster and uses less memory. The use of the confidence counter improves the prediction accuracy of state and Markov predictors.

Next step could be the test of a hybrid predictor which uses different techniques in parallel. A selector within the hybrid predictor selects the estimated best prediction among these different predictors. With the hybrid predictor the advantages of the different methods can be joined. A further question is: can the confidence estimation also improve the prediction accuracy of the other approaches. In this paper we considered only next location prediction. A further investigation should be to expand the techniques to predict locations at a certain time in future.

References

1. Daniel Ashbrook and Thad Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7(5):275–286, 2003.
2. Amiya Bhattacharya and Sajal K. Das. LeZi-Update: An Information-Theoretic Framework for Personal Mobility Tracking in PCS Networks. *Wireless Networks*, 8:121–135, 2002.
3. I-Cheng K. Chen, John T. Coffey, and Trevor N. Mudge. Analysis of Branch Prediction via Data Compression. In *ASPLOS VII*, pages 128–137, Cambridge, Massachusetts, USA, October 1996.
4. Karthik Gopalratnam and Diane J. Cook. Active LeZi: An Incremental Parsing Algorithm for Sequential Prediction. In *Sixteenth International Florida Artificial Intelligence Research Society Conference*, pages 38–42, St. Augustine, Florida, USA, May 2003.
5. Khomkrit Kaowthumrong, John Lebsack, and Richard Han. Automated Selection of the Active Device in Interactive Multi-Device Smart Spaces. In *Workshop at UbiComp'02: Supporting Spontaneous Interaction in Ubiquitous Computing Settings*, Gothenburg, Sweden, 2002.
6. Rene Mayrhofer. An Architecture for Context Prediction. In *Advances in Pervasive Computing*. Austrian Computer Society (OCG), April 2004.
7. Michael C. Mozer. The Neural Network House: An Environment that Adapts to its Inhabitants. In *AAAI Spring Symposium on Intelligent Environments*, pages 110–114, Menlo Park, CA, USA, 1998.
8. Donald J. Patterson, Lin Liao, Dieter Fox, and Henry Kautz. Inferring High-Level Behavior from Low-Level Sensors. In *5th International Conference on Ubiquitous Computing*, pages 73–89, Seattle, WA, USA, 2003.
9. Jan Petzold. Augsburg Indoor Location Tracking Benchmarks. Context Database, Institute of Pervasive Computing, University of Linz, Austria. http://www.soft.uni-linz.ac.at/Research/Context_Database/index.php, January 2005.
10. Jan Petzold, Faruk Bagci, Wolfgang Trumler, and Theo Ungerer. Global and Local Context Prediction. In *Artificial Intelligence in Mobile Systems 2003 (AIMS 2003)*, Seattle, WA, USA, October 2003.
11. Jan Petzold, Faruk Bagci, Wolfgang Trumler, and Theo Ungerer. Confidence Estimation of the State Predictor Method. In *2nd European Symposium on Ambient Intelligence*, pages 375–386, Eindhoven, The Netherlands, November 2004.
12. Jan Petzold, Andreas Pietzowski, Faruk Bagci, Wolfgang Trumler, and Theo Ungerer. Prediction of Indoor Movements Using Bayesian Networks. In *Location- and Context-Awareness (LoCA 2005)*, Oberpfaffenhofen, Germany, May 2005.
13. Sheldon M. Ross. *Introduction to Probability Models*. Academic Press, 1985.
14. Wolfgang Trumler, Faruk Bagci, Jan Petzold, and Theo Ungerer. Smart Doorplate. In *First International Conference on Appliance Design (1AD)*, Bristol, GB, May 2003. Reprinted in Pers Ubiquit Comput (2003) 7: 221-226.
15. Lucian Vintan, Arpad Gellert, Jan Petzold, and Theo Ungerer. Person Movement Prediction Using Neural Networks. In *First Workshop on Modeling and Retrieval of Context*, Ulm, Germany, September 2004.